NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

SURFSKED AN OPTIMIZATION AID FOR SURFACE COMBATANT INTER-DEPLOYMENT SCHEDULING

by

Vern F. Wing

September 1986

Thesis Advisor: R. Kevin Wood

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution is unlimited
2b DECLASSIFICATION / DOWNGRADING SCHEDULE	
4 PERFORMING ORGANIZATION REPORT NUMBER(S)	5 MONITORING ORGANIZATION REPORT NUMBER(S)
6a NAME OF PERFORMING ORGANIZATION: Naval Postgraduate School; 6b OFFICE SYMBOL (If applicable): Code 55	7a NAME OF MONITORING ORGANIZATION: Naval Postgraduate School
6c ADDRESS (City, State, and ZIP Code): Monterey, California 93943-5000	7b ADDRESS (City, State, and ZIP Code): Monterey, California 93943-5000
8a NAME OF FUNDING/SPONSORING ORGANIZATION; 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO; PROJECT NO; TASK NO; WORK UNIT ACCESSION NO

11 TITLE (Include Security Classification)
SURFSKED AN OPTIMIZATION AID FOR SURFACE COMBATANT INTER-DEPLOYMENT SCHEDULING

12 PERSONAL AUTHOR(S)
Wing, Vern F.

13a TYPE OF REPORT	13b TIME COVERED	14 DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Master's Thesis	FROM ____ TO ____	1986, September	78

16 SUPPLEMENTARY NOTATION

17 COSATI CODES			18 SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	SURFSKED; Linear-Integer Optimization; Column Generation

19 ABSTRACT (Continue on reverse if necessary and identify by block number)

The surface force inter-deployment scheduling process is the means by which units of the U.S. Navy are slated to accomplish maintenance, training, and inspection events in preparation for planned deployments or emergent missions. The schedule objective is to maximize fleet readiness while meeting the constraints of fuel, budget, home port time, and availability of supporting services.

This study provides a computerized model (SURFSKED) to assist schedulers in the optimization of the inter-deployment schedule. A set-partitioning model is used in a two-stage heuristic process to minimize scheduling costs subject to constraints on support assets.

The model is tested using a combination of actual and hypothetical data for 96 ships of the Pacific Fleet. The test runs include 88 event types and generate 13 week (one quarter) schedules.

20 DISTRIBUTION/AVAILABILITY OF ABSTRACT: ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	21 ABSTRACT SECURITY CLASSIFICATION: Unclassified
22a NAME OF RESPONSIBLE INDIVIDUAL: Prof. R. Kevin Wood	22b TELEPHONE (Include Area Code): (408) 646-2523; 22c OFFICE SYMBOL: Code 55Wd

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE

UNCLASSIFIED

Approved for public release; distribution is unlimited.

SURFSKED
An Optimization Aid for Surface Combatant Inter-Deployment Scheduling

by

Vern F. Wing
Lieutenant Commander, United States Navy
B.S., University of Washington, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 1986

Author: Vern F Wing
Vern F. Wing

Approved by: R. Kevin Wood
R. Kevin Wood, Thesis Advisor

Gerald G. Brown, Second Reader

Peter Purdue, Chairman,
Department of Operations Research

Kneale T. Marshall
Kneale T. Marshall,
Dean of Information and Policy Sciences

ABSTRACT

The surface force inter-deployment scheduling process is the means by which units of the U.S. Navy are slated to accomplish maintenance, training, and inspection events in preparation for planned deployments or emergent missions. The schedule objective is to maximize fleet readiness while meeting the constraints of fuel, budget, home port time, and availability of supporting services.

This ~~study~~ provides a computerized model (SURFSKED) to assist schedulers in the optimization of the inter-deployment schedule. A set-partitioning model is used in a two-stage heuristic process to minimize scheduling costs subject to constraints on support assets.

The model is tested using a combination of actual and hypothetical data for 96 ships of the Pacific Fleet. The test runs include 88 event types and generate 13 week (one quarter) schedules.

TABLE OF CONTENTS

ACKNOWLEDGMENTS

I wish to express my appreciation to the many people who aided in the formulation and development of this project. First, I must acknowledge the support of the COMNAVSURFPAC schedulers, especially LCDR Jim Bloom, who were supportive during the data gathering and model formulation phases and whose critique of the final product provided thoughtful insight and critical recommendations for future improvements of SURFSKED.

I am indebted to LT Clarke Goodman whose successful application of optimization methodology to ship scheduling contributed significantly to this research and to LCDR Mark Mitchell who supported me during the development of a precursor model.

I am especially thankful to my wife, Darcy, and my two children who provided continuing support and encouragement, celebrated the successes, and temporized the setbacks. SURFSKED literally could not have been developed without them.

Finally, I must express my most sincere appreciation to my thesis advisor, Professor Kevin Wood. Throughout the development, refinement, and testing phases, he was a consistent wellspring of sound advice, technical competence, professional assistance, and moral support. The quality of

this study is due largely to his insight, perseverance, and editorial effort. Without his assistance, this project could not have been undertaken.

I. INTRODUCTION

Inter-deployment scheduling is the process by which surface, air, submarine and marine units of the fleet are slated to accomplish a progression of maintenance, training, and certification events which build readiness for future operational commitments. The importance of proper scheduling is emphasized in NWP-1: "A properly balanced employment schedule is essential to attain high states of readiness, because the individual requirements for maintenance, training, and morale are frequently in competition with each other." [Ref. 1] This thesis develops and tests a computerized optimization model, specialized to surface ships of the Pacific fleet, to assist in the creation of balanced inter-deployment schedules.

The required, inter-deployment events are designed to achieve several goals:

* Enhance material condition of the units through periods of maintenance at the unit, intermediate, and shipyard levels.

* Ensure crew proficiency through formal shore-based and underway training.

* Certification of public and crew safety and crew proficiency in the operation of installed equipment and systems.

* Provide adequate home port time between operational periods in order to enhance morale.

* Conduct those inspections and certifications mandated by public law.

* Support the intra-type and intra-service training requirements of submarine, air, and marine forces.

Present scheduling is accomplished without significant computer assistance, relies heavily upon heuristics, and is therefore necessarily manpower-intensive. To produce an optimal "balanced employment schedule," which promotes total force readiness, all possible schedules would have to be examined, at least implicitly, and the one which best promoted fleet readiness while minimizing conflict between the above goals would be chosen. In order to examine all possible schedules, a high-speed computer should be utilized, scheduling rules and priorities formally quantified and a coherent measure of effectiveness developed. The model and computational methods developed in this paper seek to meet exactly these criteria.

Given the number of variables and permutations involved, it is unlikely that a schedule produced through the present manual process is as good as it might be. It is almost certainly not optimal in any objective sense. (In view of the number of possibilities, it is noteworthy that feasible schedules are developed at all.) Given the large number of contingencies which inevitably occur after schedule promulgation, changes in the remaining schedule are frequently necessary. The frequency of the rescheduling effort makes an even stronger case for use of a computerized, optimizing scheduling aid.

The main criteria which an inter-deployment schedule must satisfy are "attainability" and "feasibility." A schedule is defined to be attainable if the ship can complete, in the time allotted, all events to which it has been assigned. A schedule is unattainable if, for example, events which must occur in a specific order are out of order or the spacing between pairs of related events is insufficient. Feasibility means that the ships' schedules, in aggregate, must remain within the constraints imposed by the assets of the supporting commands. Beyond satisfying the above criteria, a feasible aggregate schedule should consist of a set of "good" attainable schedules. For example, a schedule's tempo should neither over- nor under-task a unit.

The SURFSKED model, proposed and tested in this paper, is designed to reduce the inter-deployment scheduling problem into a coherent, solvable form and to act as an aid to the human schedulers. It seeks to maximize force benefit of the schedule by minimizing deviations from ideal schedules while observing constraints of fuel, operating tempo (OPTEMPO), and support service availability. Additionally, it accounts for differences in event "needs" caused by ship type and class as well as by schedule cycle. Further, it observes the constraints imposed by event duration and periodicity, prerequisites, compatibility, and spacing. It accounts for the changing priority a ship has as it gets closer to deployment and allows for events or sequences of events to be "locked" into the schedule it creates for any combination of ships and events.

A. PROBLEM SCOPE AND DEFINITION OF TERMS

The operational component of the U.S. Pacific Fleet consists of surface, air, submarine, and marine units and their associated staffs. The focus of this paper is on scheduling for the surface element of the fleet although the methods developed here may be extended to other areas.

Ships assigned to the Pacific Fleet are assigned to the operational control of the numbered fleet commanders. Typically, ships rotate from assignment to the Seventh Fleet and operations in the Western Pacific and Indian Ocean back to assignment with the Third Fleet for operations in and around their respective home ports. The time spent in the Third Fleet is devoted primarily to upkeep, training, and certification tasks while preparing for another operational assignment to the Seventh Fleet. For purposes of this paper, the period that a ship spends preparing for deployment to the Seventh Fleet is the "inter-deployment" or "work-up" period. The "inter-deployment cycle" refers to the specific type of work-up period which is contingent on what the ship will do upon completion of work-up and how it entered the period. Thus, the inter-deployment cycle for a given ship may be from regular overhaul to deployment, deployment to deployment, etc.

Ships can be assigned to the Seventh Fleet individually but, more frequently, they enter and leave the inter-deployment cycle in groups. For example, seven to ten ships may accompany an aircraft carrier as part of a carrier battle group (CVBG), a group of surface combatants may form an SCTG (Surface Combatant Task Group) or a BBTG (Battleship Task Group) or amphibious units may form an ARG (Amphibious Ready Group). Further, ships may enter the cycle at different times and depart simultaneously or vice versa. Group arrivals and departures imply that ship schedules must be as nearly synchronized as support constraints allow which is but one special complication that must be accounted for in the scheduling process.

Ships are divided into "types" by their main mission, i.e., Guided Missile Cruiser, Destroyer, Oiler, Amphibious Landing Dock, etc. Types of ships are further differentiated by their "class." Thus, there exist TICONDEROGA, LEAHY, BELKNAP, etc., classes of Guided Missile Cruisers. Classes can contain one or more ships. The events which must be scheduled for a given ship during the inter-deployment cycle are a function of both the ship type and its class.

The "events" which comprise the schedule can be classified according to the following criteria:

* Major employment or concurrent event
* Training, maintenance, certification, etc.

* Inter-service or intra-service
* Supported or independent, i.e., conducted by outside trainers, inspectors, etc., or utilizing only unit assets
* Underway or inport
* Duration of event
* Prerequisite events, if any.

Many complications of the scheduling process are caused by the nature of the events themselves. Certain events must be scheduled alone while others may allow or require concurrent scheduling. Some events have prerequisite events and some must be repeated several times during a work-up cycle. Both the duration and periodicity of events vary with the particular event and sometimes vary by the class and/or type of ship. Additionally, some events which are notionally different require the same assets or services. All of these interrelations must be captured in a feasible schedule.

The events that a ship must accomplish during the inter-deployment period depend on the ship's cycle; how it entered the work-up period, and what it will do upon completion of the cycle. Ships can enter the cycle in one of three ways:

* Completion of a deployment
* Completion of a regular overhaul
* Completion of builders' trials (i.e., new construction).

Ships also leave the cycle in three ways:

* Commencement of a deployment
* Commencement of a regular overhaul
* Decommissioning.

The cyclic differences are captured in scheduling templates which serve as the guide for schedule formulation. COMNAVSURFPAC OPORDER 201 [Ref. 2] contains "Modular Scheduling Templates" by type and class of ship which contain ideal characterizations of the major inter-deployment events. To understand the scope of the scheduling problem, some knowledge of the current practice which translates these ideal schedule templates into operational requirements is useful.

B. CURRENT PROCEDURES

The scheduling of ships' activities can be divided into two distinct areas--long-range planning (out to five years) and short-range planning which extends from the present for about one year.

The long-range schedule provides sketchy operational information but, in general, provides start and stop dates for major events such as regular overhauls, selected restricted availabilities, deployment windows, and activation or deactivation dates for ships slated to enter or leave the force. This schedule is highly tentative and is updated continuously as changes occur or more detail can be added.

The short-range schedule consists of four quarters; the present (current operating quarter) and the first, second, and third "out quarters." The schedule for the operating quarter accounts for every day and for every ship. Only scheduling outlines exist for the out quarters.

The first "out quarter" is also called the "planning quarter." The first step in transforming the scheduling outline for this quarter into a detailed schedule is taken at the unit level. Each command independently formulates a tentative schedule based on the appropriate template. While each command knows precisely what it must accomplish and the time frame in which it must be completed, it does not know, with any degree of certainty, the needs of other ships which are competing for the same supporting resources nor does it know the schedules of the commands which will be required to support their proposed schedules.

Once formulated, these schedule proposals are submitted up the operational chain-of-command. Each successive layer reviews the proposed schedules and seeks to refine them through integration with other proposals.

Finally, a quarterly scheduling conference is convened, at the Fleet level, at which staff representatives from group level and above and all supporting commands are present. This conference lasts for approximately one week and produces a detailed listing of the "planning quarter" as well as more detailed outlines of the new "out quarters."

At the conference, supply and demand are integrated for the ships, the supporting commands, and for inter- and intra-type services. The resulting schedule may not (and probably will not) resemble the proposals submitted by the individual units. It may, in fact, vary considerably from the modular scheduling template for any individual ship. These deviations are principally caused by:

* The conflict between the requirements of individual units versus the priority given to CVBG/SCTG/ARG groups.
* Inter- and intra-type services requested versus those available.
* The high priority requirements for near-term deployers to complete remaining inter-deployment requirements in order to meet firm deployment or other operational commitments.

The present process suffers from the following problem. While schedules proposed by each unit are feasible in that they represent a command's best judgment of attainability, they may not be feasible in aggregate due to supply constraints of supporting commands. As the proposed schedules move up the chain-of-command and are reviewed and revised to maintain supply feasibility, they may lose attainability at the unit level. Admittedly, every effort is made to preserve attainability but, the vast number of possible permutations far exceeds human schedulers' capabilities to investigate more than a few. For example a single ship which requires ten events has 10! (over 3.6 million) permutations in which those events can be arranged. If some additional concurrent events are needed, the number will be even larger. When multiplied by the number of ships in the force, an astronomical number of possibilities exist. In arriving at a schedule, the present process utilizes past experience, templates, and numerous heuristics to reduce the number of possibilities. The lack of objective, quantified criteria is a noteworthy major weakness of the present system. Throughout the entire process, computers are used only in a data storage and retrieval role, not as decision aids. As a result, the schedules produced are arguably feasible but, most probably, are not optimal.

Thus, a need exists for a computerized aid to assist human schedulers. A scheduling aid need not discriminate at the daily level of detail. A weekly "time-step" will produce sufficient detail, optimize selection of the sequence of major events, and permit human schedulers to add finer detail and/or refine the schedules produced by the scheduling aid.

Once constituted, changes to the present quarter schedule occur virtually daily. From accidents to lack of availability of supporting assets, from emergent operational commitments to factors which delay the start or completion of scheduled events; environmental changes force schedule changes. Furthermore, changes may be necessitated by the fact that the promulgated schedule, while feasible on paper, is simply not attainable by the ships themselves. For example, it may over-task a given unit and thereby set the stage for changes caused by that unit's inability to accomplish all slated events. Because of event inter-relationships, a change to one ship's schedule frequently necessitates changes to other ships' schedules.

Figure 1.1 illustrates the present process. One possible cause of the problems with this process is that the amount of information available to the decision makers increases at each step in the process. Thus, a powerful argument can be made for the initial centralization of the scheduling process where decisions can be made with all pertinent information available. By this means, schedules could be created from all possible schedules for each ship. This method would maximize <u>force</u> benefit in contrast to the present system which attempts to maintain supply feasibility while making minimum modifications to schedule proposals which are based on limited information.

C. SCHEDULING CRITERIA AND ASSUMPTIONS

Implicit in the foregoing discussion are three factors which form the foundations of the surface combatant scheduling problem.

First, due to the limited number of private and public shipyards, and the constant demand for ship repair and modernization which is beyond ships' force capability, major maintenance periods frame the schedule. These major maintenance events block out significant portions of the

Level	Action	Information Available	Feasibility/ Optimality
Unit	Schedule proposal based on unit needs and "desires"	Needs of unit and some information about supporting services. No information about desires of other units	Attainable in unit's opinion and satisfies "desires"
Squadron/ Group	Revise/integrate proposed schedules to reflect squadron/ group priorities and known support constraints	Proposed schedules for all ships in squadron/ group plus above "need" information	Enhanced supply feasibility due to integration of aggregate. May be less optimal in desirability
Scheduling conference	Revise, integrate, promulgate schedules for all ships	Above plus other type commanders' demands, all support schedules, etc.	Supply feasibility maintained at possible expense of attainability

Figure 1.1 Current Scheduling Process

fleet's scheduling and all other events must be adapted to the constraints they impose on scheduling flexibility.

The second major factor influencing the scheduling problem is the employment plan of the CV battle groups, ARG's, and SCTG's. The schedules for these groups of ships are predicated on strategic goals and treaty commitments in the Pacific and Indian Oceans. While there is considerable flexibility in the selection of the individual ships which make up these groups, once constituted, they are in virtual lock-step for work-up purposes. That is, the end date of their work-up cycle is fixed to comply with broader national goals.

Finally, a more subtle influence is exerted by type commanders (TYCOM's) of carriers and submarines. Given the strategic importance of carriers and submarines, they quite simply enjoy a higher priority for scarce resources than do the elements of the surface force. For example, if a CV is in need of a particular inspection team on a given date, history has shown that the team will not be available elsewhere on that date regardless of a surface combatant's priority, readiness, or need.

Based on the foregoing factors, the model in this paper makes the following assumptions:

* The maintenance schedule drives the rest of the scheduling problem. Therefore, all major maintenance events have known start and stop dates.

* The composition of all deploying groups of ships and the deployment start and stop dates are known. (Goodman

[Ref. 3] develops a means of optimizing deployment scheduling.)

* The needs of the other TYCOM's are known in advance and the availability of supporting assets are decremented accordingly. (Other TYCOM's could use this model first to determine when those intra-type services could best be scheduled.)

These three assumptions determine the "boundary conditions" of the schedule by fixing when individual ships will enter and leave the cycle, by indicating those parts of the schedule which are predetermined, and by specifying which supporting assets will remain to meet the demands of the surface force.

Once the boundary conditions are known, the success of a scheduling aid will depend on how factors which influence event selection and timing are identified and quantified. The factors accounted for in SURFSKED are outlined below and described in detail in Chapter II.

* Priority--a ship's relative priority as compared to other ships in the inter-deployment cycle
* Need--the events needed by each ship
* Supply--the amount, timing, and availability of supporting assets
* Major vs. Concurrent--whether an event is a major employment or a concurrent event
* Compatibility--which major events are compatible with a given concurrent event
* Schedule Lock-Ins--whether normal scheduling is preempted by the existence of "locked-in" events
* Prerequisites--whether events have prerequisites and, if so, whether they have been satisfied
* Spacing--the inter-event timing of related events

* OPTEMPO/PERSTEMPO--the amount of underway and away-from-home port time contained in each schedule, respectively

* Event Duration--accounts for event-to-event variations in duration

* Time-Distance--to insure sequential events allow sufficient transit time

D. ADVANTAGES OF COMPUTER-ASSISTED SCHEDULING

Because of the nature of the factors which affect the decision process, it is possible to capture their essence in computer code. The problem can then be formulated to optimize the benefit received by the whole force. The question then, is not <u>if</u> but <u>when</u> the process should be computerized. Some of the advantages of a computerized scheduling aid have already been stated. A more complete list includes:

* Reduce the manpower intensive tasks associated with the scheduling process.

* Generate schedules which maximize <u>force</u> benefits.

* Consider <u>all</u> feasible solutions and generate the "best" which will allow decision makers to focus on individual problem areas.

* Normalize and standardize the scheduling process consistent with stated goals and supply constraints.

* Allow analysis of binding constraints in order to focus attention on where support services need to be increased or may be decreased.

* Allow multiple run analysis to determine if support service schedules are supporting the schedule or driving it.

* Save money and time currently expended on the creation of suboptimal schedules.

In summary, SURFSKED produces a thirteen-week schedule divided into one-week increments. It presupposes use as an aid to schedulers, not a replacement for them. While it is the task of a scheduling conference to produce schedules for the planning quarter whose resolution accounts for each day for each ship, no practical scheduling aid needs to produce schedules which are this "fine-grained." The purpose of SURFSKED is to optimize timing of important events among all possible permutations, and within imposed constraints, which will allow human schedulers to concentrate on important details and schedule refinements.

E. THESIS OUTLINE

This study presents a method for computerizing the surface combatant inter-deployment scheduling problem. SURFSKED utilizes a set-partitioning formulation applied to 96 surface combatants of the Pacific Fleet. Because this thesis is meant to be used by Naval schedulers who may not be versed in mathematical programming, the basics of the model and the solution procedures are developed without mathematical programming concepts in Chapter II. Chapter III presents the set-partitioning formulation of the scheduling problem and gives details of the generator which creates tentative schedules for each ship.

Finally, Chapter IV contains test results, conclusions, and recommendations based on use of SURFSKED.

Ultimately, the importance of this thesis will be determined by whether or not it, or some other computerized aid, is incorporated into the real-world scheduling process. The day that a scheduling aid is developed and implemented will be hastened by the widest dissemination of the knowledge that a means exists to computerize the problem. For this reason, SURFSKED has been tested on hypothetical fleet data: this maintains the model on an unclassified basis. The process by which the fleet input data were generated is explained in Appendix C.

In summary, this thesis is designed to acquaint both the technically oriented and the practitioner with a base-line procedure for surface combatant scheduling which has the potential to revolutionize the fleet scheduling process.

II. SOLUTION METHODOLOGY

The goal of SURFSKED is to create an optimal quarterly schedule, at a weekly level of detail, for all ships in the inter-deployment cycle. As demonstrated in Chapter I, the needs of each ship in the cycle and the schedule constraints are well defined. This chapter explains the basic solution methodology employed in the model.

A. BASIC SOLUTION METHODOLOGY

In order to solve the inter-deployment scheduling problem, three basic steps will be taken.

* Generate all attainable candidate schedules.
* Evaluate each schedule produced and assign it a cost which depends on how far it deviates from an ideal schedule.
* Select one schedule for each ship, from the set generated above, to create an overall fleet schedule. The combination of selected schedules must minimize total cost without violating constraints imposed by supporting assets.

The third step, schedule selection, is a difficult combinatorial problem whose development will be left until Chapter III. The criteria used in schedule generation and the scheme used for cost evaluation are developed below.

B. SCHEDULE GENERATION

As a base-line case, assume that each ship in the cycle must complete exactly thirteen one-week events during the quarter. Further, assume that no prerequisites exist and that no events may be scheduled concurrently. Schedule generation is then "reduced" to forming all $13! \cong 6.2 \times 10^9$ permutations of those thirteen events. Obviously, it would be impractical to generate this number of candidate schedules for even a single ship much less for each ship in the entire fleet. Fortunately, the real-world complications involved in scheduling such as event durations (longer than one week), scheduling "windows," i.e., periods during which events should be scheduled and event precedence dramatically reduce the number of schedules which must be generated. Using only needed events, the basic methodology of schedule generation is:

* Start with a major event and check to see if any concurrent events can be added.

* Continue adding major/concurrent events to the partial schedule until it is at least 13 weeks long.

* Print the completed schedule.

* Whenever a partial schedule cannot be completed, or a schedule is completed, "deschedule" the last event and try to complete the resulting partial schedule as above using other events.

* Repeat until all attainable schedules have been created.

The following sections detail the criteria which affect the scheduling decision, explain the rationale for including each in the model, and describe the means by which they were included in the model formulation.

The definitions below enable analysis of the steps taken to include the decision criteria:

INDICES:

$i = 1,\ldots,I$	Ship index where I is the total number of ships to be scheduled
$j = 1,\ldots,J$	Event index where J is the total number of events on the event list
$k = 1,\ldots,K$	Week index where K is nominally thirteen, the number of weeks in the schedule.

DATA:

NDD_i — The next deployment start date for ship i

S_{ij} — The "state-of-need" for ship i where

$$S_{ij} = \begin{cases} 1 & \text{if event } j \text{ is needed by ship } i \\ 0 & \text{otherwise} \end{cases}$$

R_{ij} — The event "requirements" for ship i expressed in weeks required

$$R_{ij} = \begin{cases} n & \text{if event } j \text{ is needed by ship } i \\ 0 & \text{otherwise} \end{cases}$$

and n = the number of weeks needed

PER_j — The inter-event period for event j

DUR_j — The duration of event j in weeks

SUP_{jk} — The "supply" of the assets available to support event j in week k

$MAJFLAG_j$ — An indicator which describes event j

$$MAJFLAG_j = \begin{cases} 1 & \text{if event } j \text{ is a major employment} \\ 0 & \text{if event } j \text{ is a concurrent event} \end{cases}$$

$COMPAT_{jj'}$ — An array which indicates the "compatibility" of the events j and j'. Indicates that events j and j' may be scheduled together vice must be.

$$COMPAT_{jj'} = \begin{cases} 1 & \text{if events } j \text{ and } j' \text{ are compatible} \\ 0 & \text{otherwise} \end{cases}$$

LI_{ik} — An array which delineates schedule "lock-ins" for the scheduling quarter

$$LI_{ik} = \begin{cases} j & \text{if event } j \text{ is to be locked in for ship } i \text{ in week } k \\ 0 & \text{otherwise} \end{cases}$$

$PREQ_j$ — A list for each event which describes whether event j has prerequisites

$$PREQ_j = \begin{cases} n & \text{if there are } n > 0 \text{ prerequisites} \\ 0 & \text{otherwise} \end{cases}$$

and for each $n > 0$ a list of the events $j_1, j_2, \ldots, j_n$ which are prerequisites for event j

LCD_{ij} — The "last-completion-date" (week) of event j by ship i

VARIABLE:

SKEDWK — The week number in the scheduling quarter

Given the above definitions, the factors which affect event selection and timing are incorporated as follows.

1. Need

This factor has two dimensions which influence the scheduling process.

First, the requirements that a particular ship needs to fulfill are based on the type and class of ship. Thus, a CG16 class guided missile cruiser has a different set of events it must accomplish than an amphibious unit such as an

LST. Further, the needs of a ship are determined by how it entered the inter-deployment cycle and how it will leave it. For example, if one DD963 class destroyer enters the cycle by completing a deployment and another by completing overhaul, they will have similar, though different, requirements during the work-up, even if they are to deploy simultaneously.

This component of need is embodied in the two data matrices, S and R, in SURFSKED.

The S matrix reflects the State of need for the entire force for all events in the event syllabus, Appendix A.

Since some events may be partially completed in week 1 of a quarter, or event duration may vary by ship type/class, the R matrix (for Requirements) captures information similar to that in the S matrix but expresses it as the number of weeks of a given event yet to be scheduled.

The second dimension of need is time-based. That is, events have either implicit or explicit periodicities associated with them (e.g., once every 18 months or once per work-up cycle). Thus, a ship which completes a given event has some period, say a year, before it must complete it again. In a sense, this dimension can be considered as the "readiness" of a ship for a given event. Thus, a ship which completed an annual requirement 11 months ago is more ready to do it again than a ship which completed it 9 months ago even though both must complete it prior to deployment.

This dimension of need is captured in the penalty function, which assesses schedule costs and is described in the next section.

The readiness cost of a ship is best (lowest) within 10% of the ideal event separation. It increases if more than 110% of the period has lapsed between successive accomplishments or if less than 90% of the period has expired. The justification for increasing costs as smaller portions of the event period lapse is that scheduling events significantly before expiration of period is inefficient, i.e., an event would have to be scheduled more frequently, thus consuming more resources, etc.

2. Supply

Many events require active outside assistance for accomplishment. Shore-based trainers, nuclear weapons certification teams, and intermediate level maintenance are examples. That these support assets are in short supply relative to fleet needs is a given. Since such constraints exist, they must be accounted for in the schedule and since the availability may vary over the scheduling period, it must be accounted for week by week throughout the scheduling period.

This aspect of the scheduling problem is captured through the use of the SUP matrix data. It is used to constrain the problem as described in Chapter III. Supply availability affects generation too. If supply equals 0 in some week, no associated event may be scheduled then.

3. Major vs. Concurrent

A number of events preclude concurrent scheduling. For example, an event which requires a ship to be under way cannot be accomplished simultaneously with one which requires the ship to be in port. Similarly, some events require "whole-ship" participation and cannot be scheduled concurrently with specialized team training. On the other hand, many events can only be scheduled concurrently. This aspect is embodied in SURFSKED in two ways. The former aspect is captured in the $MAJFLAG_j$ data which describes an event as a major or concurrent employment. The latter aspect is embodied in the $COMPAT_{jj'}$ matrix. This matrix indicates the pair-wise compatibility for all pairs of events j and j'.

4. Schedule Lock-Ins

Some events are simply "locked in place" months in advance or by policy, for example major maintenance periods, deployments, commissionings and decommissionings. Flexibility is incorporated into the model by allowing any combination of events to be locked in for any combination of ships. This feature allows schedule production to incorporate hand-written, high priority schedules.

Events are locked in using data array LI. If an event is locked in, only those schedules which contain the proper event sequence and timing will be produced.

5. Prerequisites

Many events must be accomplished sequentially. An example is the engineering qualification program which consists of Mobile Training Team visits I and II followed by an Operational Propulsion Plant Exam (OPPE). A ship which needs to complete the engineering qualification must complete each of the events in the specified order. Any other ordering is nonsense.

Prerequisites are handled through the data array called PREQ. The PREQ array indicates whether an event has prerequisites and if so, what they are. Thus, when attempting to schedule a given event, the PREQ array is consulted for a list of prerequisite events and then the S matrix is consulted to see if the prerequisites are satisfied.

6. Spacing

In addition to the prerequisite problem above, schedules must allow sufficient time between related events for lessons learned in a prerequisite event to be put into force and practiced prior to scheduling a follow-on event.

This criterion is met through the use of three parallel arrays: SEPR gives the ideal inter-event separation; SLACK defines a "window" around the ideal separation; and PENA lists the severity of the penalty function for scheduling events outside of their respective windows. These three arrays inhibit the creation of schedules which attempt to accomplish too much or too little. This criterion is also included in the cost function and is described in detail in the next section.

7. Duration

A particular event may not be able to be scheduled due to its duration. For example an event which requires two weeks to accomplish may not be scheduled one week before a "locked-in" event. This may seem obvious, but it does limit the number of possible schedules. This facet of the scheduling problem is dealt with through the creation of the DUR array which lists the nominal duration for each event. Flexibility is built into the model to vary the duration for different ships through the R matrix explained above.

8. Time-Distance

The laws of physics must be observed in scheduling. Thus sequential events in a schedule must allow sufficient time for a ship to get from the location of the first event to the location of a subsequent event.

In general, this factor is accounted for in the "definition" given to an event in the event syllabus and in the supporting data structures explained above. Test runs of SUR.SKED utilized ships whose home ports are San Diego and Long Beach and which have immediate access to the

Southern California (SOCAL) operating areas (OPAREAS). For a unit based in Pearl Harbor (or elsewhere) the event duration would be defined to include transit time if the event was to be conducted in SOCAL. Goodman [Ref. 3] made excellent use of this flexibility and clearly demonstrated its validity. Each of the above factors influences the decision process and restricts the number of feasible schedules that must be generated. Arriving at a feasible and achievable schedule will require that tradeoffs be made.

In order to prevent generation of all permutations, many of which would be patently ridiculous schedules, SURFSKED employs the above data structures to implement the following column reduction techniques.

* LOCKED-IN EVENTS--If an event is "locked-in" only those schedules which contain the event(s) in the proper weeks will be generated.

* SUPPLY LIMITATION--If no supporting supply is available in a given week, no schedule will be developed which includes that event during the restricted week.

* PREREQUISITES--If an event has prerequisites, it will not appear in a "possible" schedule until its prerequisites have been scheduled.

* SCHEDULING WINDOW--If an event's earliest ideal schedule is greater than the scheduling week being considered, it will not be scheduled.

By this means, all feasible schedules are generated recursively in a manner which allows implicit examination of all possible permutations and generation of only the feasible options. It now remains to explain how candidate schedules are evaluated once they are generated.

C. SCHEDULE COST COMPUTATION

In order to apply an optimization strategy to the scheduling problem, a means must be developed to differentiate good candidate schedules from poor ones. The process of evaluating candidate schedules must capture the essence of real-world concerns, be consistent, and it must provide sufficient discrimination.

In general, when monetary terms fail to adequately describe "costs," penalty functions are utilized to "price out" options. In their usual form, penalty functions measure deviation from ideal criteria and assign costs which are proportional to the deviation. Usually, as in the case of the surface combatant scheduling problem, penalty functions must be developed for each separate factor which influences the decision process and total cost of an option is determined by a combination of terms.

As developed, SURFSKED accounts for the costs associated with four distinct factors.

* TEMPO costs which account for both fuel imposed OPTEMPO considerations and morale imposed PERSTEMPO factors

* READINESS costs which account for the desirability of scheduling an event at a given point during the "period" of the event and the relative priority of the individual ship

* INTER-EVENT SEQUENCING (IES) costs which account for the desired separation between events

* DELETION (DEL) costs which account for the benefit lost by not scheduling other needed events.

In SURFSKED, the cost of a candidate schedule is defined to be the logarithm of the product of these factors. In its present form, SURFSKED balances the relative weights of these four factors but the relative weight given to each term is properly a variable to allow policymakers the capability to alter their importance in cost computation.

Zeleny [Ref. 4] describes the use of Multi Attribute Utility Theory (MAUT) to achieve a better fit between evaluation of options and decisionmaker preferences. MAUT supports cost functions of the form used in SURFSKED, and future research should apply MAUT to calibrate the cost function to SURFSKED's users.

The functional forms of the penalty functions utilized in SURFSKED are natural but arbitrary. They were developed to punish deviations from ideal schedules. Other functional forms, such as absolute deviation, could be used.

As with schedule generation, the analysis of cost computation first requires the definition of terms utilized in the process.

INDICES:

$i = 1,\ldots,I$	Ship index where I is the total number of ships to be scheduled
$j = 1,\ldots,J$	Event index where J is the total number of events
$k = 1,\ldots,K$	Week index where K is nominally thirteen, the number of weeks in a quarterly schedule.

DATA:

PER_j	The inter-event period for event j
PERSTEMPO	The percentage of time a ship spends in its home port between deployments

$$\text{PERSTEMPO} = \frac{\text{number of weeks away from home port since last deployment}}{\text{number of weeks since last deployment}}$$

OPTEMPO	The fraction of underway time per quarter

$$\text{OPTEMPO} = \frac{\text{number of weeks underway}}{13}$$

$READ_{ij}$	The "readiness" of ship i for event j

$$READ_{ij} = \frac{\text{time between scheduled accomplishments of event j by ship i}}{PER_j}$$

PRI_{ik}	The priority for ship i in week k of the schedule
IMP_j	The "importance" of event j

$$IMP_j = \begin{cases} 1 & \text{if deployment cannot be conducted prior to completion of j} \\ .5 & \text{otherwise} \end{cases}$$

$SEPR_{jj'}$	An array which lists the ideal "separation" in weeks between events j and j'
$SLACK_{jj'}$	The amount of deviation allowed in the separation between events j and j'
$PENA_{jj'}$	The severity of the penalty function for exceeding inter-event separation by more than the allowed deviation
$SKEDSEP_{jj'}$	The separation between events j and j' in the proposed schedule

With these terms defined, the factors in the cost function are computed as follows.

1. OPTEMPO-PERSTEMPO

The operating tempo (OPTEMPO) of a ship is the percentage of underway time a ship has in a given period of time. If the OPTEMPO is too low, readiness may suffer. If scheduled OPTEMPO is too high morale of the crew may suffer and at the extreme the schedule may simply not be attainable. Present policy prescribes approximately 27 operating days per quarter yielding an ideal OPTEMPO target of 31% in order to keep fuel consumption within allocated levels.

PERSTEMPO is defined to be the percentage of time a ship spends in its home port between deployments. A fleet goal of at least 50% is the current policy.

These two factors are evaluated as follows:

$$\text{PERSTEMPO} = \frac{\text{Number of weeks out of home port}}{\text{Number of weeks since last deployment}}$$

OPTEMPO is the fraction of under way time per quarter.

$$\text{OPTEMPO} = \frac{\text{Number of scheduled under way weeks in quarter}}{13}$$

Thus, the costs attributable to TEMPO considerations may be defined as follows:

$$\text{TEMPO}_p = \begin{cases} 1 & \text{if PERSTEMPO} \geq .5 \\ C_1 \cdot (\text{PERSTEMPO}-.5)^2 + 1 & \text{if PERSTEMPO} < .5 \end{cases}$$

$$\text{TEMPO}_O = \begin{cases} C_2(\text{OPTEMPO}-.31)^2 + 1 & \text{if OPTEMPO} > .31 \\ C_3(.31-\text{OPTEMPO})^2 + 1 & \text{if OPTEMPO} \leq .31 \end{cases}$$

$$TEMPO = (TEMPO_p \quad TEMPO_O)^{C_4}$$

The constants (C_1, C_2, C_3, C_4) are included in the formulation to allow policymakers to determine factor weight. For example, by setting $C_2 = 1$ and $C_3 = 2$ a policymaker could indicate that overemployment of a ship should be twice as expensive as underemployment. Similarly, C_1 allows adjustment between the weights given to the two individual factors ($TEMPO_p$ and $TEMPO_o$) and C_4 allows adjustment of the total TEMPO term in relation to the other cost factors. Generalized cost constants were utilized throughout the cost computation process to focus attention of policymakers on their importance and to illustrate the ease with which factor weights may be varied.

2. Readiness Costs

Readiness costs are a measure of how "ready" a particular ship is to conduct a given event and the relative scheduling priority enjoyed by that ship.

As a ship's deployment date approaches, the criticality of assigning the remaining events it must complete prior to deployment increases. Thus, a ship with less than three months before deployment has a higher scheduling priority than one which is just returning from deployment which may have 12 or more months to prepare for extended operational commitments. Priority is time-based and is a function of the deployment date of the individual ship.

SURFSKED incorporates the priority of a ship as follows.

$$PRI_{ik} = \begin{cases} 1 & \text{if } NDD_i - SKEDWK \le 13 \\ 2 & \text{if } 13 < NDD_i - SKEDWK \le 26 \\ 3 & \text{if } 26 < NDD_i - SKEDWK \le 39 \\ 4 & \text{if } 39 < NDD_i - SKEDWK \le 52 \\ 5 & \text{if } 52 < NDD_i - SKEDWK \end{cases}$$

Thus, the scheduling priority increases incrementally as a function of nearness of deployment.

The readiness costs of a schedule are a function of three factors; ship priority, readiness to accomplish an event, and the relative importance of accomplishing a particular event. SURFSKED captures these factors as follows.

$$COST_R = \sum_{k=1}^{13} PRI_{ik} \times \begin{cases} C_5 \times (READ_{ij} - 1.1)^2 \times IMP_j + 1 \\ \quad \text{if } READ_{ij} > 1.1 \\ C_6 \times (.9 - READ_{ij})^2 \times IMP_j + 1 \\ \quad \text{if } READ_{ij} < .9 \\ 1 \quad \text{if } .9 \le READ_{ij} \le 1.1 \end{cases}$$

$$\text{READINESS COST} = (COST_R)^{C_7}$$

3. <u>Inter-Event Spacing Costs</u>

Good schedules allow sufficient time between related events for lessons learned in prerequisite events to be put into practice. Thus, another measure of the "goodness" of a candidate schedule will be a function of the deviation from ideal inter-event spacing (IES). This deviation is incorporated in the penalty function below.

$$
COST_{(IES)} = \begin{cases} C_8 \times PENA_{jj'} \times (SKEDSEP_{jj'} - SEPR_{jj'})^2 + 1 \\ \quad \text{if } SKEDSEP_{jj'} > SEPR_{jj'} + SLACK_{jj'} \\ C_9 \times PENA_{jj'} \times (SEPR_{jj'} - SKEDSEP_{jj'})^2 + 1 \\ \quad \text{if } SKEDSEP_{jj'} < SEPR_{jj'} - SLACK_{jj'} \\ 1 \quad \text{otherwise} \end{cases}
$$

$$
PENALTY_{(IES)} = (COST_{(IES)})^{C_{10}}
$$

4. Deletion Costs

Given a finite supply of supporting assets and a finite (i.e., thirteen-week) scheduling horizon, it is quite possible that all event requirements of a particular ship cannot be accomplished in the scheduling quarter. The important point is that the "best" schedule for a given ship will contain the events most needed and will defer accomplishment of lower priority events to out-quarters. Implicit in this criterion is the availability of sufficient time prior to deployment to accomplish those events which are deferred.

Thus, deletion costs are incurred by leaving events out of the proposed schedule and are a function of three factors: how much time the ship has left before deployment, where the event falls on the readiness cost curve, and the importance of the event. Then, for all events j that are needed but not included in the proposed schedule, define:

$$TIMING_j = \frac{13 - LCD_j}{PER_j}$$

Then,

$$COST_{(DEL)} = 1 + (5-PRI_{13}) \times \begin{cases} \sum_{\substack{j \in \text{Needed} \\ \text{but not} \\ \text{scheduled}}} IMP_j \times TIMING_j & \text{if } TIMING_j > 1.1 \\ 1 & \text{if } TIMING_j < 1.1 \end{cases}$$

$$PENALTY_{(DEL)} = (COST_{(DEL)})^{C_{11}}$$

The total cost, $COST_T$, is defined as the product of these factors.

$$COST_T = TEMPO \cdot READINESS\ COST \cdot PENALTY_{(IES)} \cdot PENALTY_{(DEL)}$$

The aggregate schedule cost is considered to be the product of individual ship schedule costs. To effect this in an additive set partitioning model, $\log_{10}(COST_T)$ is used in the objective function. Thus, if $COST_T$ is the cost of the nth schedule, $C_n = \log_{10}(COST_T)$.

SURFSKED, as tested, has "balanced" the weight of cost factors by assessing the mean value of each factor on a sampling run and weighting the constants to achieve balance among the means. Policymakers may change the relative weights without loss of generality in the model.

In formulating the data which support both the generation and evaluation functions of SURFSKED, scheduling templates [Ref. 3] and Naval standards for OPTEMPO and PERSTEMPO were utilized. Any candidate schedule can be evaluated in terms of deviations from the ideal using this data. A perfect schedule yields a total cost of 0 while less ideal schedules produce costs which are higher.

It should be noted that while the generator adheres to the scheduling rules stated in the previous section, it will generate schedules which may deviate significantly from the ideal. However, the evaluator assigns high costs to those schedules which deviate significantly from the norm. Thus, any attainable schedule is permitted in the final solution but at a cost inversely proportionate to its quality.

Formal cost evaluation offers advantages over the present scheduling practice. First, objective schedule evaluation permits the application of optimization theory. Alone, this would be insufficient cause to convert from the present "paper-and-pencil" scheduling method. However, the following advantages, even when viewed in isolation from the power of the rest of the methodology, are themselves sufficient justification to pursue optimization technology.

* The process identifies specific (objective) criteria which differentiate good schedules from poor ones.

* Use of objective criteria minimizes impact of changes in personnel (experience factor) due to personnel rotation.

* The process standardizes the measure of quality and normalizes it across the force.

* Creation of objective criteria involves the policy-makers and permits systematic review and revision of parameters as goals or constraints change.

* Penalty functions, once constructed, can be "calibrated" through Multi Attribute Utility Theory to realistically capture the priorities of policymakers.

Once all candidate schedules have been generated and their costs evaluated, the solution to the scheduling problem is to select exactly one schedule for each ship such that the set of selected schedules minimizes total costs without violating the supply constraints of supporting assets. The formulation of the problem as a set-partitioning model which accomplishes this goal is the subject of the next chapter.

III. MODEL DEVELOPMENT AND DESCRIPTION

Set-covering, set-packing and set-partitioning methods have been used for many years as a means to solve certain types of scheduling problems (see Bausch [Ref. 5]). While the basic formulation is straightforward, it often proves impractical on large-scale problems due to the number of variables generated in this type of formulation and the limitations of most general purpose optimization software. In order to solve such problems efficiently, a special purpose, large-scale solver is necessary. The X-System [Ref. 6], developed by Brown and Graves, is an advanced optimization routine which enables the efficient solution of large-scale integer and mixed-integer problems. This package employs several sophisticated techniques (hyper-sparse data representation, elastic programming, etc.) in order to solve large problems in a reasonable amount of computer time. The system has been successfully used on problems involving thousands of variables and constraints. Goodman, for example, employed the X-System on an Atlantic Fleet scheduling problem involving over ten thousand variables and over two hundred constraints [Ref. 3].

The flexibility that the set-partitioning approach provides is an especially important benefit when viewed in the context of the surface combatant scheduling problem.

Due to the binary nature of the decision of what to schedule and when to schedule it, and the large number of permutations involved, the problem is particularly well suited to this type of formulation. When coupled with the power of the X-System, a set-partitioning approach provides a very efficient means of solving the inter-deployment scheduling problem.

A. THE SURFSKED SET PARTITIONING FORMULATION

The SURFSKED "set-partitioning" model is described below. In fact, this is a generalized set-partitioning model since it contains inequality constraints in addition to equality constraints and because the right-hand-side values $b_{F(j,k)}$ are general integers, not necessarily 1's.

The SURFSKED model is:

1. Indices:

$i = 1,...,I$	Ship index where the total number of ships being scheduled is I.
$j = 1,...,J$	Event index where the total number of possible events which can be scheduled is J.
$k = 1,...,13$	Week index where k represents the k^{th} week of a 13 week quarterly schedule
$n = 1,...,N$	Column index where the formulation has N columns

$$F(j,k) = I + k\,((j-1) \cdot 13), \quad k = 1,...,13;\ j = 1,...,J$$

2. Data:

c_n, $n = 1,...,N$ Cost of schedule n

$$a_{in} = \begin{cases} 1 & \text{if schedule } n \text{ is for ship } i \\ 0 & \text{otherwise} \end{cases}$$

$$a_{F(j,k)n} = \begin{cases} 1 & \text{if schedule } n \text{ requires asset } j \text{ in week } k \\ 0 & \text{otherwise} \end{cases}$$

$b_{F(j,k)}$ = the supply of j available in week k. k = 1,...,13 (the number of weeks in a quarterly schedule).

3. Decision variable:

$$x_n = \begin{cases} 1 & \text{if schedule } n \text{ is selected} \\ 0 & \text{otherwise} \end{cases}$$

4. Formulation:

$$\min \sum_{n=1}^{N} c_n x_n$$

$$\text{s.t.} \quad \sum_{n=1}^{N} a_{in} x_n = 1 \,, \qquad i = 1,\ldots,I \qquad (1)$$

$$\sum_{n=1}^{N} a_{F(j,k)n} x_n \le b_{F(j,k)} \,, \qquad j = 1,\ldots,J,\ k = 1,\ldots,13 \qquad (2)$$

$$x_n \quad 0,1\}$$

Constraints (1) are "ship-schedule" constraints and require that exactly one schedule per ship be selected. Constraints (2) ensure that the available resources, i.e., supporting assets, are not exceeded for any event in any week.

Figure 3.1 provides a matrix representation of a SURFSKED formulation where the number of ships in the "force" equals three, the number of events is four, and a scheduling horizon of two weeks is used. The solution demands that at most one column (i.e., one schedule) be selected for each ship and that the total cost to the force be minimized. In this simple example, it can be seen that setting x_1, x_5, and x_9 equal to 1, with all other decision variables equal to 0, provides a "force" schedule with cost equal to 7. This is the optimal solution. While this example can be solved by inspection, the real-world case requires the use of a sophisticated solver like the X-System.

In the example problem depicted in Figure 3.1, the sense of the inequalities for the supporting asset supply constraints imply that services are being provided to the surface force. For example, rows 4 and 5, event 1, may imply that two inspection teams are available in each week and therefore the number of ships that may be scheduled for event 1 in each week is limited to two. However, the surface force is also the provider of intra-type services such as DLQ/HIFR, carrier escort (plane guard), NGFS spotter

		S_1				S_2			S_3				
		X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9			
SHIPS	1	1	1	1	1	0	0	0	0	0	=	1	Schedule Constraints
	2	0	0	0	0	1	1	1	0	0	=	1	
	3	0	0	0	0	0	0	0	1	1	=	1	
EVENT 1	WK 1	1	0	0	0	0	1	1	1	1	$\leq$	2	
	WK 2	0	0	0	1	1	0	0	0	0	$\leq$	2	Supporting Asset Supply Constraints
EVENT 2	WK 1	0	1	0	0	0	0	0	0	0	$\leq$	1	
	WK 2	0	0	1	0	0	0	1	1	0	$\leq$	1	
EVENT 3	WK 1	0	0	0	1	0	0	0	0	0	$\leq$	1	
	WK 2	1	1	0	0	0	1	0	0	0	$\leq$	1	
EVENT 4	WK 1	0	0	1	0	1	0	0	0	0	$\leq$	3	
	WK 2	0	0	0	0	0	0	0	0	1	$\leq$	2	
COSTS		C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9			
		3	5	3	7	1	1	6	5	3			
		*				*				*			

* - indicates columns in optimal solution

Figure 3.1. Matrix Representation of SURFSKED Set-Partitioning Formulation

training, and submarine surface target services (see Appendix A). In this case, the right hand side would indicate the number of required surface combatants and the "$\leq$" would be changed to either "=" to imply an exact numerical requirement, or to "$\geq$" to imply a lower bounded requirement.

Ultimately, the practicality of this formulation depends on two criteria; the ability to generate all feasible columns (schedules) for each ship, and the ability to assign "costs" to each column generated. Costs were examined in Chapter II; the remaining sections of this chapter deal with column generation and reduction.

B. COLUMN GENERATION

The needs of any ship are easily established by the boundary conditions of deployments and major maintenance events, the ship type and class, and the specific work-up cycle for the individual ship. Once these needs are known, there are a finite, but probably still large, number of permutations of the needed events which can possibly fill a thirteen-week schedule. The essence of SURFSKED is that it implicitly examines each of the possible permutations and generates only those candidate schedules that are attainable.

The following algorithm will generate all attainable schedules for a ship, where only major events are considered.

DEFINE:

E	A list of needed events; $e_1, e_2, \ldots, e_n$
S	A stack of events taken from E
e_j; $j = 1, \ldots, n$	A needed event where n represents the total number of needed events
L(S)	Length of partial schedule in S
algorithm	Generate;

```
  input:  E  a list of needed events
  output: S  a schedule of events with length ≥ 13

begin
  S = Ø
  j1 = 1
Next; for j = j1 to n
      begin
        if [ATTAINABLE(S,ej)]
        begin
          S ← S + ej
          if  L(S)  ≥ 13
          begin
            print S
            S ← S - ej
            j1 = j + 1
          end
          else j1 = 1
        end
      end
      if S = Ø halt}
      S ← S - ek   /*ek is top element in S*/
      j1 = k+1
      go to Next
end
```

The function ATTAINABLE(S,e_j) returns "TRUE" if e_j can be added to the partial schedule S without violating attainability criteria; it returns "FALSE" otherwise. The function checks that:

* the first event added to S is a major event.
* lock-in events are scheduled in their proper sequence and at the locked-in time.

* support assets are available in the proposed scheduling week

* prerequisite events (if any) have been completed

* the proposed event is within its scheduling window

Slight modifications of the algorithm allow handling of concurrent events. In practice, SURFSKED sorts E into prerequisite order such that all events appear on the list after their prerequisite events. As a result, generator speed was improved by approximately 3000%. While this sort is not specifically an attainability check (nor is it necessary for successful generation), an increase in generator efficiency is realized by elimination of partial schedules which lead to unattainability prior to reaching the desired schedule length.

In summary, SURFSKED uses a recursive process to generate the candidate schedules which form the columns of the A matrix. Validity of the generated columns is guaranteed through the successive attainability checks. All permutations of events which create a schedule of at least thirteen-weeks duration are implicitly examined and those that are attainable are explicitly evaluated and added to the list of candidate schedules. Finally, the X-System solver is used to select the specific combination of schedules for the entire force such that each ship has exactly one schedule, total cost is optimized, and supply constraints remain inviolate.

C. SUPPORTING DATA

The amount of data which must be entered is considerable. Fortunately, the majority of it is "one-time data base construction" and the remainder could be accumulated in "real-time" if SURFSKED were fully implemented.

* ONE-TIME DATA--Data in this category are event descriptors (prerequisites, period, duration, major/concurrent codes, under way factors, importance, inter-event compatibility, separation, slack, and penalties, etc.) and ship type/class need vectors.

* ACCUMULATED DATA--Once implemented, the system could track historical completion dates.

* PRE-RUN DATA ENTRY--This requires manual entry for "locked-in" events such as deployment start/stop dates, major maintenance periods and the supply availability for the scheduling quarter.

Clearly, the amount of data entry (after system implementation) is modest and will certainly be less time consuming, and therefore less expensive than the current process.

IV. TEST RESULTS AND CONCLUSIONS

The SURFSKED model was developed, implemented, and tested at the Naval Postgraduate School on an IBM 3033 AP computer operating under the CMS operating system. Test runs were conducted using 96 ships of the Pacific Fleet. The event syllabus (Appendix A) contains 88 events of which 55 were "major employments" and 33 were "concurrent events." Schedules of 13-week duration were constructed at a one week level of discrimination. Sample output, in Gannt chart (line diagram) format, is contained in Appendix B.

Comparison of SURFSKED solutions against known solutions was precluded by two factors. First, extant historical scheduling data reflects what was executed by the force, not what was scheduled for execution. Secondly, current data base management "over-writes" completion dates for events each time they are completed. The first factor precludes meaningful comparison while the second deprives the model of necessary input data. In addition, tests using current real-world data would require classification of this thesis and would restrict distribution.

Thus, data used to test SURFSKED were compiled from two sources. Current (1985) scheduling directives and OPORDERS's were used to compile the data base information which describes events (i.e., duration, period, compatibility, etc.). Ship history and need data plus supply constraint data were constructed as described in Appendix C.

Model efficiency is discussed in terms of CPU time necessary for model processing. Results are summarized in Table 4.1.

A. DATA PROCESSING CONSIDERATIONS

The SURFSKED model can be conveniently divided into three functional modules: Schedule generator with imbedded evaluator, solver, and report writer.

1. Generator

The SURFSKED schedule generator/evaluator is written (in approximately 1000 lines of code) in ANSI FORTRAN 77 and compiled by the IBM VS FORTRAN compiler at OPT(3). Input data are formatted arrays which include all data-base information describing event parameters, ship need and historical data, and supply/timing data for supporting assets. Candidate schedules and problem size parameters are directed to two formatted output files which are in turn read by the solver.

2. Solver

The X-System solver is written in FORTRAN 66 and is compiled by the IBM VS FORTRAN compiler at OPT(3) and LANGLVL(66). Input data consists of the generator output files. Solver output is written to a formatted file which serves as input to the report writer.

3. Report Writer

The SURFSKED report writer is written in ANSI FORTRAN 77. Output is written in Gantt chart format suitable for a line printer. This format closely parallels the content and construction of the "line diagram" format used by fleet schedulers. Sample output is contained in Appendix B.

B. TEST RUN METHODOLOGY

While successfully yielding reasonable numbers of schedules, i.e., a few hundred schedules, for some ships, the reduction techniques are not sufficiently restrictive, in general. Too many attainable schedules exist for some ships. This is largely due to the concurrent events which have few prerequisites or other limitations imposed on their scheduling. Since too many schedules were being generated, a "two-stage heuristic" was implemented.

This heuristic which solves a restriction of the original problem. First, it constructs schedules with the needs of most concurrent events suppressed. Since many concurrent events are compatible with several major employments, many attainable schedules involve the same sequence of major employments and differ only in the timing of concurrent events. Yet, the majority of schedule quality is dependent on timing and selection of major events which make up a candidate schedule.

Thus, in order to limit generation of the thousands of permutations on inherently poor schedules, this method generates candidate schedules based on ship needs for the 55 major employments and the 7 concurrent events which are prerequisite to major events. The output from the generator is optimized by the solver which in turn yields 96 basic schedules.

These basic schedules are read into the lock-in matrix and the generator is again used to build permutations on only these schedules based on the ships' needs for all concurrent events. The generator output is then again optimized by the solver and printed.

The results are summarized in Table 4.1 with various limits imposed on the maximum number of schedules produced for each ship. This method is admittedly heuristic but produces schedules whose objective values appear to be approaching the lower limit; the values improve only slightly as the column limit is relaxed. The eight schedules in Appendix B are taken from the last test run and are representative of the quality of schedules produced.

Other solution strategies are also possible but have not been explored in this study. Either dynamic cost evaluation and limiting could be employed or dynamic column generation could be utilized e.g., [Ref. 7]. Dynamic cost evaluation would compute a lower bound on cost for a partial schedule

TABLE 4.1

SUMMARY OF RESULTS USING TWO-STEP METHOD

First Step			
column limit/ship	200	200	300
# rows	1240	1240	1240
# columns	5240	5240	7149
# non-zeros	76,811	76,811	105,483
generator CPU time	13.0	13.0	16.5
solver CPU time	22.4	22.4	28.1
objective function value	53.29	53.29	52.44
Second Step			
column limit/ship	100	200	200
# rows	1240	1240	1240
# columns	4043	7217	7317
#non-zeros	85,386	158,711	158,930
generator CPU time	11.5	18.4	19.0
solver CPU time	18.9	30.6	30.9
objective function value	61.01	59.77	59.41

and terminate generation of any schedules exceeding a certain limit. Dynamic column generation would first solve a restricted problem, i.e., a problem with only a subset of all attainable schedules. Then it would create new schedules, but only those which could improve the overall objective function value. The point is--strategies do exist which will enable application of optimization techniques on problems which exceed solver and/or generator capacity.

C. RECOMMENDATIONS

While SURFSKED demonstrates the feasibility of optimizing surface combatant inter-deployment scheduling through a set-partitioning formulation, room exists to refine the model.

One area which requires further research is the cost evaluation function. Zeleny [Ref. 4] suggests a method by which multi-attribute utility theory (MAUT) may be applied to decisions involving multiple criteria. MAUT techniques extend the accuracy of the log-product formulation by tailoring results to the decisionmaker's preferences. Since schedule cost is used to determine the optimal force schedule, it is imperative that the penalty function mimics real world policy preference as closely as possible. SURFSKED, as formulated, is only coarsely calibrated. The finalization of penalty function functional forms, determination of weighting constants, and application of MAUT techniques represent an additional thesis-level research task.

A second area which deserves further study was suggested by the schedulers of the COMNAVSURFPAC staff. As presently formulated, SURFSKED uses the scheduling templates contained in COMNAVSURFPAC OPORDER 201 [Ref. 2] as the ideal when evaluating candidate schedule deviation. It has been suggested that since each individual ship is in the best position to evaluate unit needs, and event timing, that unit schedule proposals be utilized as the reference ideal. The SURFSKED formulation can be altered to accommodate this refinement. In fact, use of unit-level schedule proposals as the evaluation standard will reduce the extent of data base construction necessary to support SURFSKED.

Another area of possible research has already been mentioned: dynamic evaluation and dynamic generation techniques. Through these means, partial schedules could be evaluated as they are constructed, resulting in early termination of partial trees (schedules) which will lead to poor complete schedules. Successful application will dramatically reduce the number of candidate schedules produced by reducing the number of permutations of inherently poor schedules that are generated.

Finally, generator efficiency and selectivity could be improved. The imbedded attainability checks are extensive but are not exhaustive. Further attainability checks may be identified through close contact with end users. The payoff for this effort will be in increased generator efficiency, as well as solutions which are closer to optimal. If sufficient additional checks can be identified and implemented, problem size reduction strategies may not be necessary and a truly optimal solution could be obtained.

D. OTHER APPLICATIONS

SURFSKED was formulated to address the surface combatant inter-deployment scheduling problem, but the methodology may be extended to other scheduling problems as well. By changing the event syllabus and constructing the appropriate data base, the method and the model can be used by submarine, air, and marine units.

E. CONCLUSION

SURFSKED is not yet an end user product. It is a "proof-of-concept" which demonstrates that high quality schedules can be constructed automatically and with great efficiency. It demonstrates that the inter-deployment scheduling problem can be reduced to coherent form, that scheduling rules and priorities can be quantified and standardized, and that the goal of constructing optimal or near-optimal, balanced fleet schedules is attainable.

Further, SURFSKED demonstrates the applicability of the set-partitioning approach to the large-scale inter-deployment scheduling problem. The flexibility of the method allows incorporation of all currently identified scheduling criteria and has the potential to accommodate future refinements. While this study only touches upon the issue of support constraint analysis, SURFSKED's usefulness as a tool in this analysis may ultimately prove to be of equal importance to fleet schedulers.

Finally, while SURFSKED is not yet a finished product, it establishes a base-line model that was wholly Navy developed to meet a Navy need. The facilities, faculty, and students of the Naval Postgraduate School have the capability to produce a final product. Realization of a viable scheduling aid depends only on sponsorship of continuing research by an end-user command.

APPENDIX A

EVENT SYLLABUS

Event #	Schedule Acronym	Meaning
1	ROH	Regular overhaul
2	TAV	Tender availability
3	SRA	Selected repair availability
4	IMAV	Intermediate maintenance availability
5	PREOVHL:UPK	Pre-overhaul upkeep
6	UPK	Upkeep
7	HOLUPK	Holiday upkeep
8	LVUPK	Leave and upkeep
9	RFS	Ready for sea
10	RAV	Restricted availability
11	WSAT/CSSQT	Weapons' systems acceptance trials/Combat systems ship's qualification trials
12	MATINSP	Material inspection
13	POTANDI	Pre-overhaul test and inspection
14	PSA	Post shakedown availability
15	IMAUPK	Intermediate maintenance availability upkeep
16	IMAV:SIMA s/s	Ship-to-shop intermediate maintenance availability
17	TAV:s/s	Ship-to-shop tender availability
18	ADINSP:ASI	Annual supply inspection

19	3M ASSIST VST	3M assist visit
20	ADINSP:3M	3M inspection
21	CSRT	Combat system readiness trials
22	HARPCERT	HARPOON certification
23	TOMCERT	TOMAHAWK certification
24	ADINSP:MEDINSP	Medical inspection
25	INSURV	Board of inspection and survey
26	HRAV	Human resources availability
27	LOE-GT	Light-off-exam, gas turbine
28	LOE-STM	Light-off-exam, steam
29	LOE-DIES	Light-off-exam, diesel
30	MTT1-GT	Mobile training team visit 1, gas turbine
31	MTT1-STM	Mobile training team visit 1, steam
32	MTT1-DIES	Mobile training team visit 1, diesel
33	MTT2-GT	Mobile training team visit 2, gas turbine
34	MTT2-STM	Mobile training team visit 2, steam
35	MTT2-DIES	Mobile training team visit 2, diesel
36	MTT3-STM	Mobile training team visit 3, steam
37	MTT3-DIES	Mobile training team visit 3, diesel
38	OPPE-GT	Operational propulsion plant exam, gas turbine
39	OPPE-STM	Operational propulsion plant exam, steam
40	OPPE-DIES	Operational propulsion plant exam, diesel
41	OPPRE-GT	Operational propulsion plant re-exam, gas turbine

42	OPPRE-STM	Operational propulsion plant re-exam, steam
43	OPPRE-DIES	Operational propulsion plant re-exam, diesel
44	ISE/ECC	Independent ship exercises, engineering casualty control
45	PREORSE ISE	Pre-operational reactor safeguards exam, independent ship exercise
46	MTT ORSE	Mobile training team, operational reactor safeguard exam
47	ORSE	Operational reactor safeguard exam
48	PRE OPPE UPK	Pre-operational propulsion plant exam, upkeep
49	PRE ORSE UPK	Pre-operational reactor safeguard exam, upkeep
50	TYTIPT:9037	Nuclear weapons administrative assist
51	TYTIPT:90X4	Nuclear weapons administrative assist
52	NWAT	Nuclear weapons acceptance training
53	NWAI/DNSI	Nuclear weapons acceptance inspection/Defense nuclear surety inspection
54	NGFS TNG VST	Naval gunfire support training visit
55	NGFS:SCI	Naval gunfire support San Clemente Island
56	LOAD:SBCH	Ammunition onload, Seal Beach
57	OFLD:SBCH	Ammunition offload, Seal Beach
58	LOAD:NORIS	Ammunition onload, North Island
59	OFLD:NORIS	Ammunition offload, North Island
60	TYTIPT:TMA	Target motion analysis training (1079)

61	TYTIPT:PAS PLOT	Pass plotting technique training (1086)
62	TYTIPT:SONO P/P	Sonobuoy passive plotting training (1081)
63	TYTIPT:ASW PH1	Phase one ASW training
64	TYTIPT:ASW PH2	Phase two ASW training
65	TYTIPT:AAW INT	AAW intermediate training (0084)
66	TYTIPT:AAW ADV	AAW advanced training (0085)
67	TYTIPT:20B4/ RAVIR	Mobile van AAW training
68	TYTIPT:HARP T/T	HARPOON team training (0122)
69	TRE	Training readiness evaluation
70	RFT1	Refresher training phase one
71	RFT2	Refresher training phase two
72	MOORFT	Modified refresher training
73	PHIBRFT	Amphibious refresher training
74	EXER:COMPTUEX	Composite training unit exercise
75	EXER:READEX	Readiness exercise
76	EXER:FLEETEX	Fleet exercise
77	EXER:LDEX	Loading exercise
78	EXER:PHIBEX	Amphibious exercise
79	VST	Port visit
80	SPECOPS	Special operations
81	ESC	Escort
82	DEPLOY	Deployed

83	POM	Pre-overseas movement
84	IPT	Inport
85	OPS:EASTPAC	Operations, Eastern Pacific
86	ASIR	Aviation safety inspection
87	HELO CERT	Helicopter certification
88	DLQ/HIFR	Deck landing qualifications/ Helicopter inflight refueling

APPENDIX B

SAMPLE SURFSKED OUTPUT

The report writer output is written in Gantt chart format which closely parallels the format and content of the "line diagrams" used by fleet schedulers. This appendix contains eight examples of the output generated during the testing phase of SURFSKED.

SKED NO. 7 SHIP NO. 2 COST 1.780 ALBERT DAVID FF1050 D WINTER 1975

OPPE | UPK | TRE | MOORFT | EXER: COMPTUEX | IPT | EXER: READEX

IMAV SIMA S/S

ADINSP 3MI

ADINSP: MEDINSP

SKED NO 628 SHIP NO 7 COST 0.349 BARBOUR COUNTY LST1195 D WINTER 1975

ISE ECC | OPPRE | IPT | UPK | IPT | PREOVHL: UPK | ROH

SKED NO 1169 SHIP NO 14 COST 1.177 CALIFORNIA CGN36 A WINTER 1975

IMAUPK | RFS | MTT ORSE | IPT | UPK | IPT | PRE ORSE UPK | PREORSE ISE

IMAV SIMA S/S

ADINSP 3MI

TYTIPT NORN | TYTIPT PAC PLUS | TYTIPT HAND P P | TYTIPT IMA 1079

SKED NO 1624 SHIP NO 18 COST 0.876 CLEAVELAND LPD7 B WINTER 1975

IMAV	RFS	IPT	ISE/ECC	MTT2	UPK		TRE	LOAD:SBCH
IMAV SIMA S.S					ADINSP:SM			
SM ASSIST VST								

SKED NO 2217 SHIP NO 21 COST 0.674 CROMMELIN FFG37 L WINTER 1975

CSRT	IMAV	RFS	EXER: COMPTUEX	IPT	EXER: READEX
	ADINSP MEDINSP				

SKED NO 6314 SHIP NO 54 COST 0.948 LEWIS B PULLER FFG23 B WINTER 1975

PRE OPPE UPK		IPT	OPPRE	TRE	MODRFT	UPK		EXER: COMPTUEX
TYTIPT COND P P	TYTIPT AAW INT					TYTIPT: ASW PH1	ASIR	

SKED NO. 8514 SHIP NO. 76 COST 0.809 RENTZ FFG46 L WINTER 1975

ISE/ECC	OPPRE	UPK	IPT	TRE	MODRFT	IPT	EXER: COMPTUEX
			TYTIPT: PAS PLOT				
			HELO CERT				

SKED NO. 9223 SHIP NO. 83 COST 0.663 SIDES FFG14 D WINTER 1975

PRE OPPE UPK	UPK	IPT	OPPRE	TRE	MODRFT	EXER: COMPTUEX
TYTIPT: ASW PH2	TYTIPT: AAW ADV				DLQ/WIFR	
HELO CERT						

APPENDIX C

METHOD USED TO CONSTRUCT HYPOTHETICAL DATA

SURFSKED requires 13 matrices as data. To avoid classification of this thesis, four of these matrices are hypothetical:

* LCD -- "Last completion date" of each event for each ship
* S -- The "state" of force "needs" expressed as a 0,1 variable for each ship and each event
* R -- The force "needs" (requirements) expressed in weeks for each ship and each event.
* LI -- The "locked-in" major events for the force

These matrices were generated using the following random, but reasonable, scheme.

First, for each ship class and type a vector of ideal next-completion-dates was constructed for each possible inter-deployment cycle (i.e., regular overhaul (ROH) to deployment, deployment to deployment, and deployment to ROH). A (discrete uniform) random number generator was employed to determine which cycle the ship was currently on and the one from which it came.

NOTE: For the FFG-7 class, vectors were constructed for post-shakedown availability (PSA) to deployment one, deployment one to deployment two, and deployment two to deployment one.

Next, ships were assigned to one of six categories.

1) In ROH (PSA for FFG-7 class ships)
2) In first quarter of work-up cycle
3) In second quarter of work-up cycle
4) In third quarter of work-up cycle
5) In fourth quarter of work-up cycle
6) Deployed

A random number generator was used to determine which category each ship was in with probability 1/6 for each category. In the case of ships in the first category, a random number was again generated to determine which quarter of ROH the ship was in. (This was not done for FFG-7 class ships.) Ships in category 6 were again randomly assigned to first-half and second-half deployment categories.

Then a uniform random integer between 1 and 13 was chosen for each ship to indicate which week the ship was in during the selected quarter. From this data, the "week-in-cycle" could be determined for each ship.

Based on the week-in-cycle, all events with a next-completion-date greater than week-in-cycle were given an S value of 1. All events in the thirteen weeks prior to week-in-cycle were given an 'S' value of 0 with probability .9 and a value of 1 with probability .1. Thus, the S matrix was made to be slightly pessimistic with respect to the deterministic, ideal next-completion-date data.

A scan was then conducted of the S matrix and the ideal next-completion-date vector and the R matrix was compiled based on indicated remaining needs for each ship.

Next, the LCD matrix was compiled by scanning backwards from the week-in-cycle date of the present cycle through the last cycle to determine a temporary "last completion date." To this matrix was added a uniform random integer drawn from the interval -3 to +3 to form the LCD matrix.

In order to simulate the scheduling demands imposed by block arrivals and departures, all ships which began or ended a deployment (as determined by week-in-cycle) in the current quarter were divided into "early" (weeks 1-6) and "late" (weeks 7-13) categories. Their deployment start and stop dates were "normalized" to the mean of the group. This last feature may have created some peculiar (though certainly possible) groupings of ships but it achieves the desired end of block arrivals and departures.

Finally, a forward scan was conducted from the week-in-cycle and the LI ("locked-in") matrix was compiled for deployment start and stop dates and major maintenance events.

In conclusion, the data used in testing of SURFSKED has a sensible randomization applied in its construction. If any fault can be attributed to the test data it is that it errs on the side of pessimism, not optimism.

LIST OF REFERENCES

1. Chief of Naval Operations Naval Warfare Publication, NWP-1, Strategic Concept of the U.S. Navy, May 1983.

2. Commander, Naval Surface Forces U.S. Pacific Fleet Operations Order 201, Current Operations.

3. Goodman, C.E., Annual Scheduling of Atlantic Fleet Naval Combatants, M.S. Thesis, Naval Postgraduate School, Monterey, California, March 1985.

4. Zeleny, M., Multiple Criteria Decision Making, McGraw Hill Book Company, 1982.

5. Bausch, D.O., Computational Advances in the Solution of Large-Scale Set Covering and Set Partitioning Problems, M.S. Thesis, Naval Postgraduate School, Monterey, California, September 1982.

6. Brown, G.G. and Graves, G.W., XS Mathematical Programming System, perpetual working paper, (c. 1983).

7. Gilmore, P.C. and Gomory, R.E., "A Linear Programming Approach to the Cutting Stock Problem," Operations Research, Vol. 9, pp. 847-859, 1961.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22304-6145	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943-5002	2
3.	Department Chairman, Code 55 Department of Operations Research Naval Postgraduate School Monterey, California 93943-5000	1
4.	Professor R. Kevin Wood, Code 55Wd Department of Operations Research Naval Postgraduate School Monterey, California 93943-5000	6
5.	Professor Gerald G. Brown, Code 55Bw Department of Operations Research Naval Postgraduate School Monterey, California 93943-5000	1
6.	Commander in Chief United States Atlantic Fleet Attn: CAPT J.C. Kronz (Code N33) Norfolk, Virginia 23511	2
7.	Commander in Chief United States Pacific Fleet Attn: Code N33 Pearl Harbor, Hawaii 96860-7000	2
8.	Commander Naval Surface Force United States Atlantic Fleet Attn: Scheduling Officer Norfolk, Virginia 23511	2
9.	Commander Naval Surface Force United States Pacific Fleet Attn: Scheduling Officer Naval Amphibious Base, Coronado San Diego, California 92155	2

10. LT Clarke E. Goodman, Jr. 1
Department Head School Class 88
Surface Warfare Officers School Command
Newport, Rhode Island 02481

11. LCDR Mark Mitchell, Code 55Mi 1
Department of Operations Research
Naval Postgraduate School
Monterey, California 93943-5000

12. LCDR Vern F. Wing 2
USS Fletcher (DD992)
FPO San Francisco, California 96665

13. Commander Third Fleet 2
ATTN: Scheduling Officer
Pearl Harbor, Hawaii 96860-7500

14. Commander Submarine Force 2
U.S. Pacific Fleet
ATTN: Scheduling Officer
Pearl Harbor, Hawaii 96860

15. Commander Naval Air Force 2
U.S. Pacific Fleet
ATTN: Scheduling Officer
Naval Air Station North Island
San Diego, California 92135-5100